



**QUEEN'S
UNIVERSITY
BELFAST**

Development of a vehicle model architecture to improve modeling flexibility

Stevens, G., Murtagh, M., Kee, R., Early, J., & Best, R. (2017). Development of a vehicle model architecture to improve modeling flexibility. *SAE International Journal of Engines*, 10(3), [2017-01-1138].
<https://doi.org/10.4271/2017-01-1138>

Published in:
SAE International Journal of Engines

Document Version:
Publisher's PDF, also known as Version of record

Queen's University Belfast - Research Portal:
[Link to publication record in Queen's University Belfast Research Portal](#)

Publisher rights
Copyright 2017 SAE.
This work is made available online in accordance with the publisher's policies. Please refer to any applicable terms of use of the publisher.

General rights
Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.



Development of a Vehicle Model Architecture to Improve Modeling Flexibility

Gary Stevens, Martin Murtagh, Robert Kee, Juliana Early, and Roy Douglas
 Queen's University Belfast

Robert Best
 Wrights Group Ltd.

ABSTRACT

In this paper a dynamic, modular, 1-D vehicle model architecture is presented which seeks to enhance modelling flexibility and can be rapidly adapted to new vehicle concepts, including hybrid configurations. Interdependencies between model sub-systems are minimized. Each subsystem of the vehicle model follows a standardized signal architecture allowing subsystems to be developed, tested and validated separately from the main model and easily reintegrated. Standard dynamic equations are used to calculate the rotational speed of the desired driveline component within each subsystem i.e. dynamic calculations are carried out with respect to the component of interest. Sample simulations are presented for isolated and integrated components to demonstrate flexibility. Two vehicle test cases are presented. The application to a conventional heavy-duty vehicle demonstrates the operational capabilities of the modelling methodology, while the inclusion of electrical components to form a mild-hybrid heavy-duty vehicle demonstrates the model's potential for predicting improvements in fuel economy and performance over a specified drive cycle. Qualitative validation characteristics are presented, highlighting the ability of the model to accurately capture dynamic events and fuel consumption profiles.

CITATION: Stevens, G., Murtagh, M., Kee, R., Early, J. et al., "Development of a Vehicle Model Architecture to Improve Modeling Flexibility," *SAE Int. J. Engines* 10(3):2017, doi:10.4271/2017-01-1138.

INTRODUCTION

The modelling of vehicle propulsion systems prior to any testing is common practice in most vehicle engineering applications. System modelling can drastically reduce both concept to production times and associated testing costs. A model allows the user to determine the advantages gained from any changes in vehicle topology and perform a cost-benefit analysis to determine whether the modifications are worthwhile before committing significant resources. Another advantage of simulation models is that they can be used to test and develop new and unproven components and systems to identify promising concepts or architectures removing the need for building prototype vehicles or test rigs saving both time and cost [1].

With rapid advancements in technology, coupled with increasingly stringent legislative requirements, there is a growing need for models which are not only reliable, but also extremely flexible. The main problem commonly encountered with model development is the lack of flexibility when the topology of the vehicle is changed significantly [2]. This paper proposes a vehicle model architecture (VMA) that attempts to address this problem. The proposed VMA attempts to build on those previously established [3,4] to improve the models flexibility and adaptability to new vehicle concepts.

Both the automotive and heavy duty sectors have migrated towards a modular modelling approach as it provides a range of advantages. It enables easily adaptable models capable of handling different

subsystems of variable complexity. For example, different engines or transmissions can be simulated in the same vehicle architecture to determine the effect on fuel consumption or any other system dynamics of interest. Libraries of components can be created and shared amongst different users which reduces development time. However, the downside of a modular modelling approach is that each component must conform to a standardized signal architecture to facilitate sub-system interchangeability. A robust VMA which is suitable in application to nearly all road vehicles has been previously described by Belton *et al.* [3]. The top-level subsystems were established based on a logical decomposition of the vehicle components. These are the driver, environment, electrical, auxiliaries, powerplant, transmission, driveline, chassis, braking, steering, vehicle system controller (VSC) and bus. Matlab/Simulink was used to develop the VMA, as shown in [Figure 1](#).

Each model subsystem shown in [Figure 1](#) contains various layers. The top layer is shown in [Figure 1](#) and connects each subsystem together through the main signal bus. This is denoted as Layer 1. Layer 2 is contained within each subsystem and illustrates how each of the required signals is transmitted into each component from the main bus and also shows how feedback is transmitted back to the main signal bus. Layer 3 then describes local subsystem controller and plant behavior. [Figure 2](#) shows how an IC engine and automatic transmission with a torque convertor would be implemented in this model architecture.

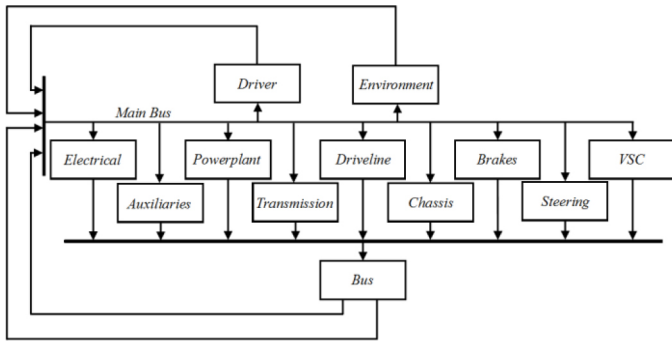


Figure 1. Top Level Vehicle Model Architecture showing critical subsystems and bus connections [3]

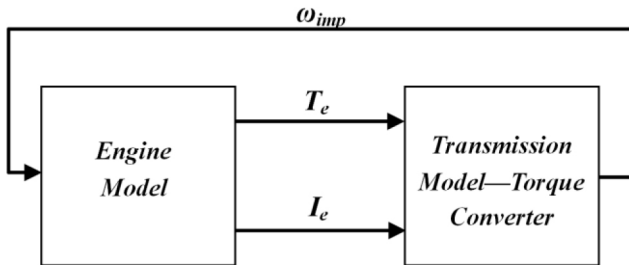


Figure 2. Engine and automatic transmission torque convertor signal paths [3]

The engine inertia and gross engine torque from the engine model are passed to the torque convertor model block. The impeller speed is then passed from the transmission subsystem back to the engine where it becomes the engine speed input for the engine model [3]. This inherently means that these two subsystems are somewhat coupled to one another as the engine model now has a level of dependency on the subsystem that is downstream of it. Further investigation was conducted on a similar model based on a similar architecture by Mason *et al.* [4]. Both compliant and non-compliant models for the same system were investigated. The non-compliant version, a schematic of which is shown in Figure 3, uses the approach of cumulative torque and cumulative inertia. These signals are carried to the end of the drivetrain system where the signals are terminated, at which point a speed parameter is calculated. This speed signal is then passed backwards through the drivetrain and transformed through gearing ratios accordingly to determine every other drivetrain components speed.

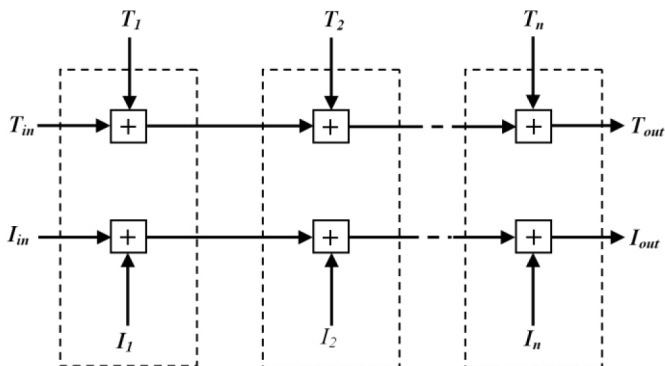


Figure 3. Modular sub models showing cumulative torque and inertia paths [4]

A similar VMA modelling approach was implemented for heavy duty vehicles by Emirler *et al.* [5]. Figure 4 shows a summarized diagram of the signal paths used in the model. In each of the model

architectures described, the subsystems involved have a level of dependency on those surrounding them. If the speed signal could be removed, allowing the speed of each component to be calculated intrinsically, this would decouple the subsystems from one another and improve the model flexibility and robustness. Other vehicle modelling packages are available, such as Advisor which uses a hybrid forward backward modelling approach. However, some users have reported limited flexibility and there can be difficulties with model reconfiguration [6]. Similarly, Mathworks uses its bespoke environment Simscape to develop its own vehicle modelling examples. However, the underlying functions of predefined components are difficult to change reducing modelling flexibility. Alternative approaches use a power flow regime, rather than torque flow, such as PSAT (Power Systems Analysis Toolbox) which is primarily used for electrical power systems [7]. This however also introduces a significant extent of coupling onto each of the subsystems in the model.

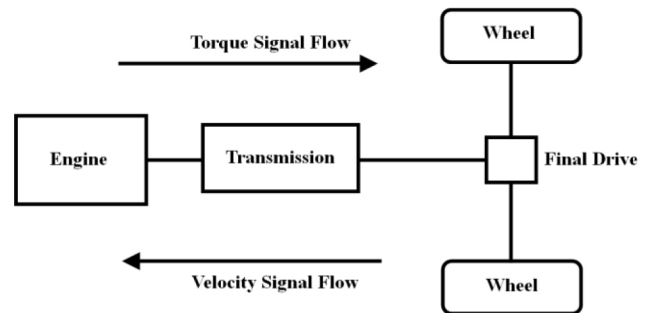


Figure 4. Rigidly connected rotational inertia system and dynamic signal flow paths as represented by modular sub models [5]

In the following sections the proposed new model architecture is established. Simulation results are then presented showing the functionality of the model, demonstrated on an engine-flywheel system, a conventional powertrain vehicle and finally a hybridized powertrain. Qualitative validation is also presented in the form of a comparison to real world driving data. This demonstrates the ability of the model to accurately capture both detailed drivetrain dynamics and fuel consumption profiles.

MODEL METHODOLOGY

The proposed VMA methodology will first be explained using simple examples for mathematical clarity. The model methodology has been approached in a dynamic, forward-facing manner to allow the implementation of existing control system models and to assist with the development of future control strategies. The intended purpose for the model developed within this current work is as a system level design tool. Thus, compliance within the system has been ignored for model simplification and all connections are assumed to be rigid, reducing the degrees of freedom of the system.

Newton's second law is used to calculate the velocity trajectory for the vehicle. The velocity of the vehicle at any instant can be found through an integration of the acceleration as shown in Equation 1:

$$v = \int \frac{\sum F_x}{m} dt \quad (1)$$

where v is the velocity in ms^{-1} , F_x is the force in N, and m is the mass in kg. In this case, the mass includes the effect of the inertia of all the powertrain components. The sum of the forces includes the applied force at the wheels, which is a function of the torque that was produced at the engine, and all of the resistive forces on the vehicle. Due to the modular modelling approach adopted, each component subsystem has its own inertia/mass programmed within it. A parameter referred to as the reduced mass moment of inertia for the system was calculated at each point, with respect to that point. This means that not only can each component model operate independently but they can also function as an integrated system without any modifications to the subsystem structure or input/output (I/O) signals. Dynamic equation parameters change depending on what is upstream or downstream of the component in terms of torque flow in the model (upstream and downstream directions being defined by torque flow from the engine to the wheels). Speed is not inferred onto any component, but intrinsically calculated. The rotational speed of any rotating component is derived by means of an integration of the resultant torque applied to the component and the reduced mass moment of inertia for the system with respect to that component as shown in Equation 2:

$$\omega = \int \frac{\sum T}{I} dt \quad (2)$$

where T is torque in Nm, I is the rotational inertia in kgm^2 and ω is the rotational speed of the component in rads^{-1} . For a single rotating mass, there is only one inertia term in the dynamic equation to consider; the inertia of the component itself. However, if two masses (I_1 and I_2) are rotating together at identical speed, then the dynamic equation which describes the motion of the system can be written as shown in Equation 3:

$$T_{app} - T_{res} = I_1 \dot{\omega} + I_2 \dot{\omega} \quad (3)$$

Where T_{app} is the applied torque on the component in Nm, T_{res} is the resistive torque on the component in Nm. Equation 3 gives a simple explanation of the term reduced mass moment of inertia if the two inertia terms are grouped, as in Equation 4:

$$T_{app} - T_{res} = (I_1 + I_2) \dot{\omega} \quad (4)$$

The term $(I_1 + I_2)$ represents the reduced mass moment of inertia for the system. For systems which contain gearing ratios between rotating components, the inertia of upstream components must be multiplied by the square of the gearing ratio when being passed downstream, and vice versa, the inertia of downstream components must be divided by the square of the gear ratio when being passed upstream. An example of a simple gear train system is shown for clarification in Figure 5.

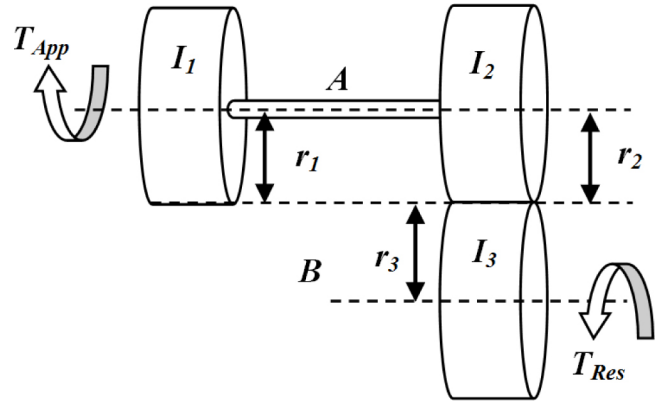


Figure 5. Simple gear train diagram showing rotating masses, rigid connections, critical dimensions and applied and resistive torques

Using this methodology, the system shown in Figure 5 can be described using two equations. One is formulated with respect to the rotating shaft A (Equation 5), and the other is written with respect to rotating shaft B (Equation 6). Note that the torque terms are also transformed when passing through gearing ratios accordingly as shown in equations 5 and 6:

$$T_{app} - \frac{T_{res}}{i} = (I_1 + I_2 + \frac{I_3}{i^2}) \dot{\omega}_A \quad (5)$$

$$T_{app} i - T_{res} = ((I_1 + I_2) i^2 + I_3) \dot{\omega}_B \quad (6)$$

Where i is the gearing ratio between the two shafts, calculated as $\frac{r_3}{r_2}$. This method allows for independent determination of the rotational speeds of both shafts via integrations of the angular accelerations. The next step is to determine how to account for the vehicle mass in the rotational inertia terms (and vice versa). Referring to the previous example, assume that the gear train shown in Figure 5 is the powertrain of a vehicle which must accelerate the vehicle mass (Inertia I_3 is effectively acting as a wheel). It can be shown that the vehicle mass is accounted for in the powertrain by multiplying by the square of the wheel radius when passing the mass upstream. Subsequently, this is combined with any other algebraic gearing ratio transformations as previously described for the system which results in a set of equations as follows (Equations 7, 8 and 9). Also the powertrain inertia must be divided by the square of the wheel radius when being passed to the vehicle model.

$$T_{app} - \frac{F_{res} r_3}{i} = \left(\left(\frac{m r_3^2 + I_3}{i^2} \right) + I_1 + I_2 \right) \dot{\omega}_A \quad (7)$$

$$T_{app} i - F_{res} r_3 = (I_3 + m r_3^2 + (I_1 + I_2) i^2) \dot{\omega}_B \quad (8)$$

$$\frac{T_{app} i}{r_3} - F_{res} = \left(\left(\frac{((I_1 + I_2) i^2 + I_3)}{r_3^2} \right) + m \right) \dot{\omega}_A \quad (9)$$

Equation 7 is the system equation written with respect to shaft A of Figure 5, Equation 8 is the system equation written with respect to shaft B and Equation 9 is the system equation written with respect to the vehicle (Note Equation 9 describes linear motion for the vehicle, whereas Equations 7 and 8 describe rotational motion for the two shafts A and B respectively). These equations again allow for independent determination of rotational speeds and vehicle speeds. This is the principle behind the mathematical decoupling of each of the subsystems from one another. The model signal architecture presented previously [3] required the feedback of a rotational speed signal from the downstream component; in that case the impeller of the torque converter was passing a speed signal back upstream to the engine to inform it of its rotational speed. The new proposed methodology passes torque and resistive torque signals, as well as calculating a reduced mass moment of inertia for the system at each point, and uses dynamic differential equations to solve for the speed intrinsically within each subsystem. This ultimately means that should a downstream component be removed; the rotational speed of the upstream component is still known. This increases the model flexibility by allowing components to be added or removed as required without adversely affecting functionality. For example, taking the previous equation which describes the rotational motion for shaft A, Equation 7, and removing both of the downstream components by setting masses m and I_3 equal to zero and removing the resistive force present at the wheel by setting F_{res} equal to zero (note that a resistive torque is included to account for any losses) results in Equation 10:

$$T_{app} - T_{res} = (I_1 + I_2)\dot{\omega}_A \quad (10)$$

This is a modified form of Equation 7 which now mirrors Equation 4 describing the motion for a shaft with two masses. This demonstrates that should a component be removed from the model, each component still behaves as it would in the same real world conditions. Equations 7, 8 and 9 can then be rearranged and integrated with respect to time to calculate the rotational speed of the shafts or the vehicle velocity from an integral with respect to time, as shown in Equations 11, 12 and 13:

$$\omega_A = \int \frac{T_{app} - \frac{F_{res}r_3}{i}}{(mr_3^2 + \frac{I_3}{i^2} + I_1 + I_2)} dt \quad (11)$$

$$\omega_B = \int \frac{T_{app}i - F_{res}r_3}{(I_3 + mr_3^2 + (I_1 + I_2)i^2)} dt \quad (12)$$

$$v = \int \frac{\frac{T_{app}i}{r_3} - F_{res}}{\left(\left(\frac{((I_1 + I_2)i^2 + I_3)}{r_3^2} \right) + m \right)} dt \quad (13)$$

Figure 6 depicts the new model signal paths relative to Figure 4. In place of a torque flow path and a speed flow path, an applied torque path and a resistive torque path now exist.

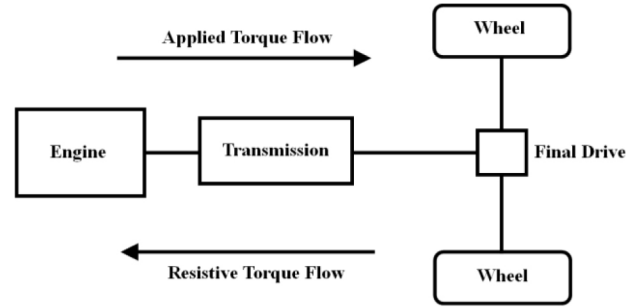


Figure 6. Rigidly connected rotational inertia system and new dynamic signal flow paths as represented by modular sub models

MODEL IMPLEMENTATION

Matlab/Simulink was used to implement the VMA which will ultimately be demonstrated on test cases using a heavy duty vehicle platform. The vehicle and powertrain were broken down into subsystems, each of which follows the signal architecture presented in Figure 7.

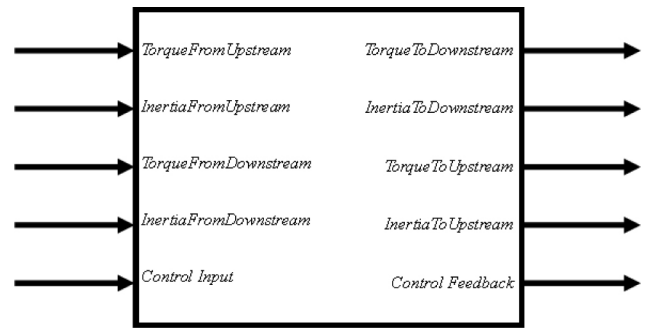


Figure 7. Subsystem signal architecture diagram of I/O

The *TorqueFromUpstream* signal is the torque applied to the component by the component in the upstream direction and is equivalent to the T_{app} parameter. The *InertiaFromUpstream* signal is the reduced mass moment of inertia of all the upstream components. The *TorqueFromDownstream* signal is the resistive torque on the component from the component in the downstream direction and is equivalent to the T_{res} parameter. The *InertiaFromDownstream* signal is the reduced mass moment of inertia of all of the downstream components. The *Control Input* signal is the generic input port for all of the control signals that the specific component requires. The *TorqueToDownstream* signal is the torque applied by the component to the component immediately downstream. The *InertiaToDownstream* signal is the reduced mass moment of inertia that is passed to the component immediately downstream. The *TorqueToUpstream* signal is the resistive torque that is passed to the component immediately upstream. The *InertiaToUpstream* signal is the reduced mass moment of inertia that is passed to the component immediately upstream. Note that all torques and inertias must be transformed accordingly through any gearing ratios within the subsystem. The components inertia is included with the inertia input signals and passed in the appropriate direction as the inertia output signals for use by other components. Similarly, any torque losses

within the component are applied to the torque input signals to give the resulting torque output signals. For clarification a simple shaft model which follows the architecture outlined is shown in Figure 8. The subsystems internal Simulink blocks are shown in Figure 9. There is no control input to this model and the shafts rotational speed (N_{Shaft}) is the control feedback. Also note that this particular model has no internal losses associated with it.

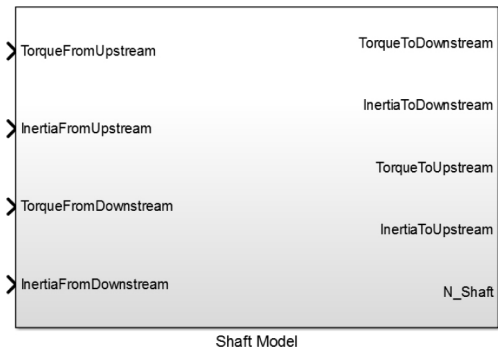


Figure 8. Simulink implementation of I/O architecture

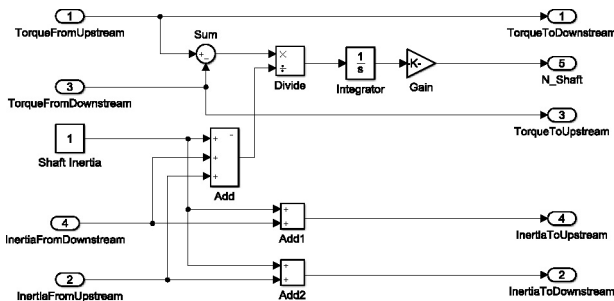


Figure 9. Simulink Block Signal Paths and equation implementation shown inside subsystem

A schematic is shown in Figure 10 depicting how two component subsystems are connected together. Each block has the same signal inputs and outputs as that shown in Figure 7. Any signal with the notation ...*ToUpstream* as a suffix means that that signal must travel to the upstream component. Similarly, any signal with the suffix ...*ToDownstream* means that that signal must travel to the downstream component. The same is also true for the signals with the ...*From...* suffix. Any signal with the suffix ...*FromUpstream* must come from the component in the upstream direction and any signal with the suffix ...*FromDownstream* must come from the component in the downstream direction.

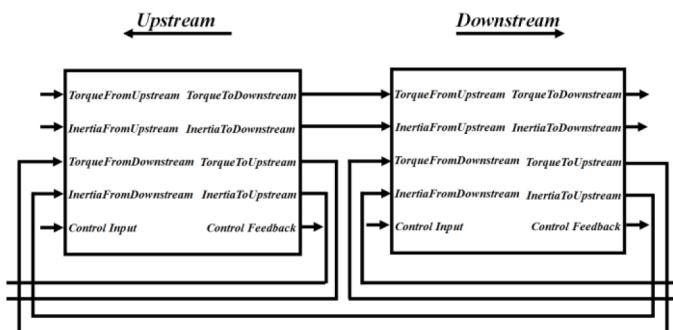


Figure 10. Two connected component sub systems

System losses are accounted for by subtracting any loss from the applied torque path and adding the loss to the resistive torque path. This means that at any point within the system, the applied torque on any component is the cumulative effect of the upstream applied torques minus any losses due to the upstream components, and the cumulative effect of the downstream resistive torques plus any losses due to the downstream components.

When hybrid components are added to the system, there can be multiple points in the drivetrain where torque and inertia are added and/or subtracted from the system. This is accounted for by adding the torque from any hybrid component to the applied torque path and subtracting it from the resistive torque path at the same point in the drivetrain. By doing so any downstream components have an increase in applied torque as expected, and upstream components perceive the torque addition as a reduction in resistive torque which can result in an overall increase in resultant torque on the upstream component, due to the sign convention used.

In the following sections the torque output from the simple engine model is a function of accelerator pedal position and the maximum available torque at any given engine speed, i.e.:

$$T_{\text{engine}} = f(APP, N_{\text{engine}}) \quad (14)$$

A motoring torque curve is used to simulate engine braking during lift off and coasting, as well as an idle governor to prevent engine stall. The cumulative resistive torque, which is a function of all the resistive forces on the vehicle and any losses within the drive train are then subtracted from the engine torque and this is used to calculate crankshaft speed.

An automatic transmission was modelled as an input shaft and output shaft coupled by gear sets of varying transmission ratio and a separate clutch for each gear set. Each clutch operates by switching states when the criteria for clutch lock up or clutch break-up are met. Clutch models have been extensively described in literature [8, 9, 10, 11, 12, 13]. Losses are included in the transmission system as torque loss lookup maps which are dependent on the current transmission operating conditions within each gear.

The final drive or axle was modelled in a similar manner to the automatic transmission but with a fixed gearing ratio between the input and output shafts and a fixed efficiency. The wheel model serves as a subsystem that converts forces to torques and inertias to masses and vice versa within the system, effectively transforming parameters from the rotational domain to the linear domain and vice versa.

The driver model is a PID controller that can be set to various levels of sensitivity by altering the gains on each the proportional, integral and derivative parameters.

SIMULATION RESULTS

Simulation results of an engine accelerating a mass are shown to demonstrate the models operational functionality. The engine simulated is designed for heavy duty applications and as such has a

relatively small operational speed range in comparison to typical automotive engines. A drivetrain consisting of an automatic transmission, final drive axle, wheels and a vehicle model is then added to the model and a comparison to real world experimental driving data is shown. Finally, hybridizing components are added to the model, in the form of torque sources that are external to the conventional drivetrain, to show the models capability in adapting to new drivetrain configurations.

Engine-Flywheel System

The input to the engine-flywheel model was the desired rotational speed for the flywheel. This was set to 2000RPM between 10 and 17.5 seconds and zero at all other times. The clutch which couples the engine and flywheel together was programmed to start its engagement at 4 seconds. The clutch was also programmed to decouple the flywheel from the engine at 20 seconds. This engagement and disengagement of the clutch causes a change in model states from 2 degrees of freedom, where the engine and flywheel rotate independently, to a single degree of freedom where the two rotate together. The driver model is based on a parallel PID controller. The output from the driver model is based on the error between the desired speed and the simulated speed. The error signal then passes through the controller to modify the accelerator pedal position to accelerate the system to the desired speed. The engine model is based on a maximum available torque curve and map based fuel consumption calculations. The clutch and flywheel system consists of a simple clutch model in conjunction with a relatively large inertia. The top level system model is shown in Figure 11. Figures 12, 13 and 14 show the engine speed and torque plots for the system. Figures 13 and 14 show the applied and resistive torque on both the engine and flywheel respectively. When the applied torque is greater than the resistive torque, the related component is accelerating and conversely, when the resistive torque is greater than the applied torque the component is decelerating.

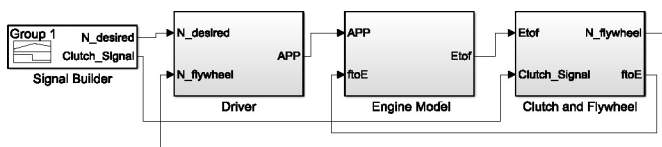


Figure 11. Engine-Flywheel system top level

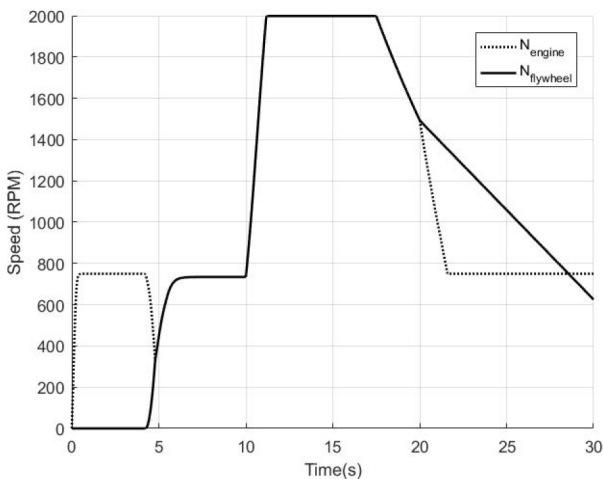


Figure 12. Engine-Flywheel system speed plot

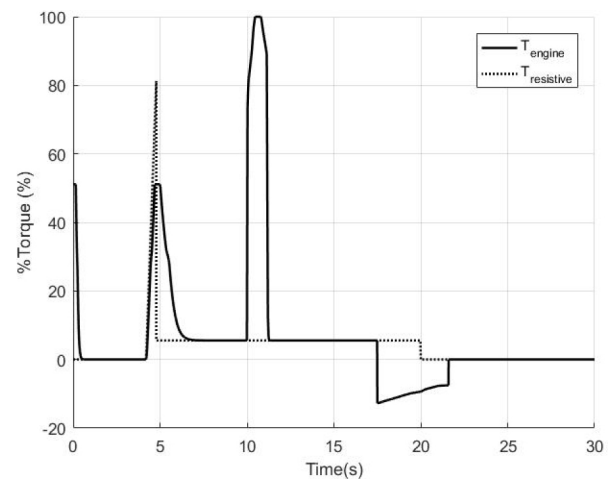


Figure 13. Engine-Flywheel system engine torque plot

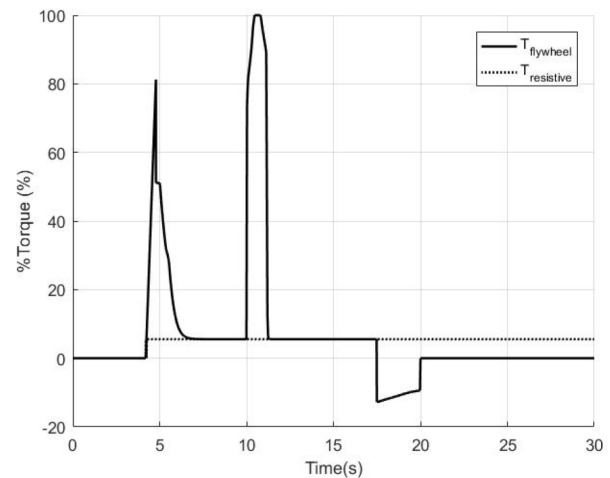


Figure 14. Engine-Flywheel system flywheel torque plot

Figure 12 shows that at the start of the simulation the engine speed increases to its idle speed. Once clutch engagement begins at 4 seconds the engine speed is pulled down and the flywheel speed increases. At the point where the engine and flywheel rotational speeds are equal the clutch is locked and the systems accelerate together back to the engine idle speed due to the idle speed governor in the model. At 10 seconds, the driver model then sees the increase in the desired rotational speed to 2000 rpm and applies additional accelerator pedal position to increase the torque output from the engine as can be seen on each of the torque plots between 10 and approximately 12 seconds. The system then continues at this desired rotational speed until 17.5 seconds in the simulation when the desired speed returns to zero. The driver model then removes its accelerator pedal output and the system begins to coast down due to a resistive torque after 17.5 seconds as can be seen in Figure 13. The clutch is then programmed to separate the systems at 20 seconds, resulting in the engine returning to its idle speed shortly after due to engine motoring torque and the flywheel continuing to coast down at a lower rate of deceleration due to its relatively larger inertia and a resistive torque due to bearing friction.

Conventional Drivetrain Vehicle

The same driver and engine combination was then used in the simulation of full vehicle-powertrain system with the drivetrain configuration that was previously described in place of the flywheel. Full load acceleration and braking events were simulated. Simulation results are shown in comparison to real world experimental driving data from a conventional drivetrain bus during testing at Millbrook Proving Ground, England. The top level system model is shown in Figure 15. The vehicle velocity plot is shown in Figure 16. Engine speed is shown in Figure 17 and engine output torque is shown in Figure 18. The input to the model was the velocity profile that the vehicle completed during real world driving. Figure 19 shows the fuel used by the vehicle during the event in the real world against the simulated results. There is a maximum difference between the two sets of fuel consumption results of 9.4% at 23.2s and an average difference of 5.2%.

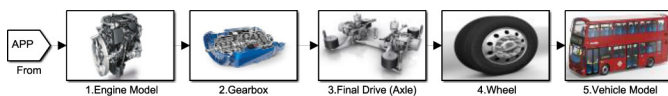


Figure 15. Conventional Drivetrain vehicle top level

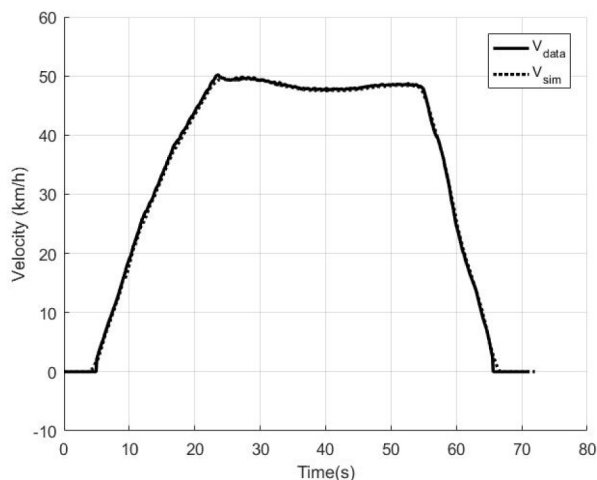


Figure 16. Conventional drivetrain vehicle velocity plot

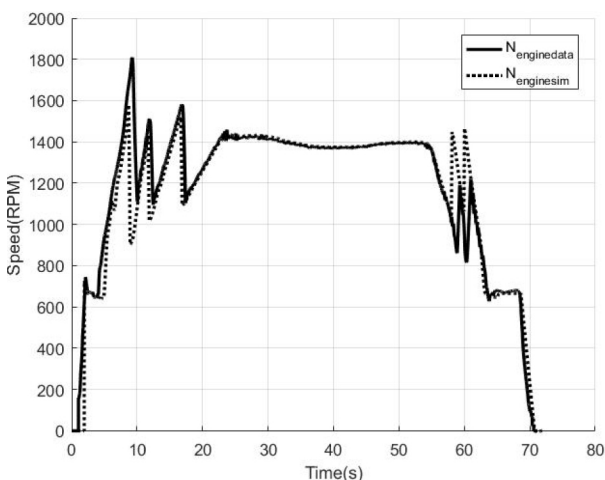


Figure 17. Conventional drivetrain engine speed plot

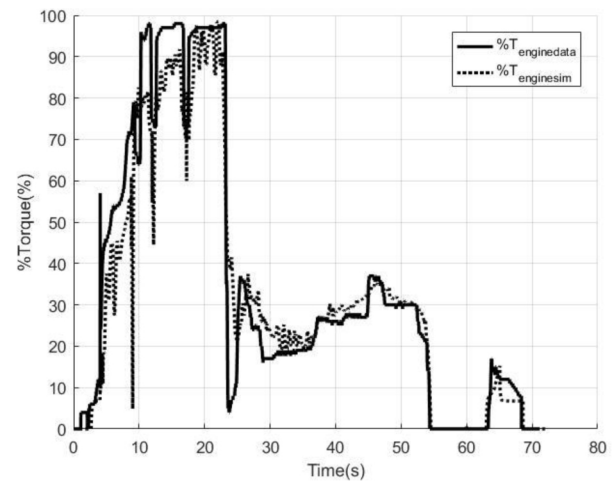


Figure 18. Conventional drivetrain engine torque plot

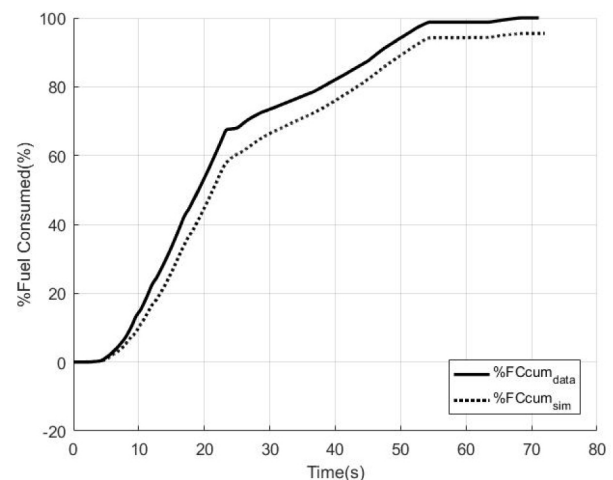


Figure 19. Fuel used during full load acceleration and deceleration event for conventional drivetrain vehicle

The simulation results show good initial correlation to the real world experimental driving data. Velocity traces match well and default gear shift points that can be seen in Figure 17 are close to those that were performed by the automatic transmission in the vehicle. Only slight discrepancies exist in the engine speed trace, indicating that the shifting strategy employed in the model needs further refinement. The model is also able to capture events such as the influence from the low speed idle governor which increases engine torque output between approximately 63 and 68 seconds. The differences in simulated and actual fuel consumption indicate that there are areas in the model which could benefit from either additional fidelity or additional refinement. For example, illustrated in the engine torque plot, Figure 18, the engine does not output the same amount of torque during the acceleration phase of the driving event. The result of this lower torque output is a lower instantaneous fuel consumption during this period. The cumulative fuel consumption is then also lower as it is calculated as an integration of the instantaneous fuel consumption. This causes the observed deviation between the predicted and actual fuel consumption profiles.

Hybrid Drivetrain Vehicle

An external torque source was included in the drivetrain, applied to the output shaft of the gearbox, to simulate the effect of a hybridizing electrical component. Shown model in Figure 20 is the top level system. A rudimentary control strategy was implemented in order to establish the effect that this component would have if assisting during the acceleration phase of the simulation. The new torque source assisted the powertrain during the acceleration phase between 10 and 20 seconds. The same full load acceleration event was repeated. The new speed and torque plots are shown in Figures 21, 22 and 23 in comparison to the previously simulated results for the conventional powertrain (i.e. non-hybrid) vehicle. The fuel consumption of the real vehicle and both simulations are plotted in Figure 24.

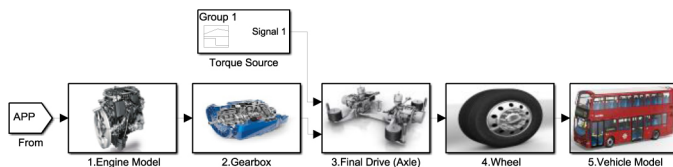


Figure 20. Hybrid drivetrain vehicle top level

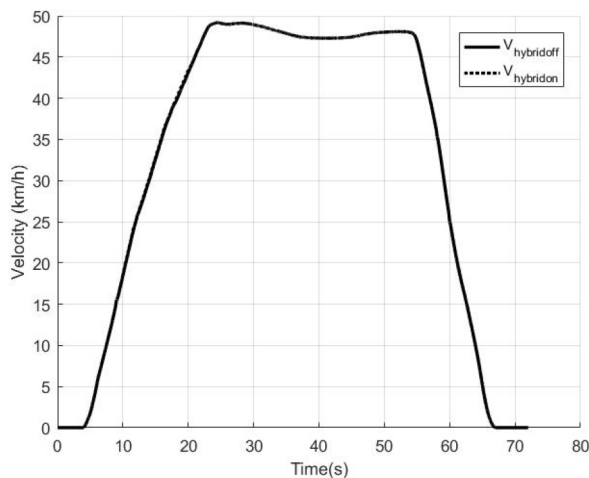


Figure 21. Hybrid drivetrain vehicle velocity plot

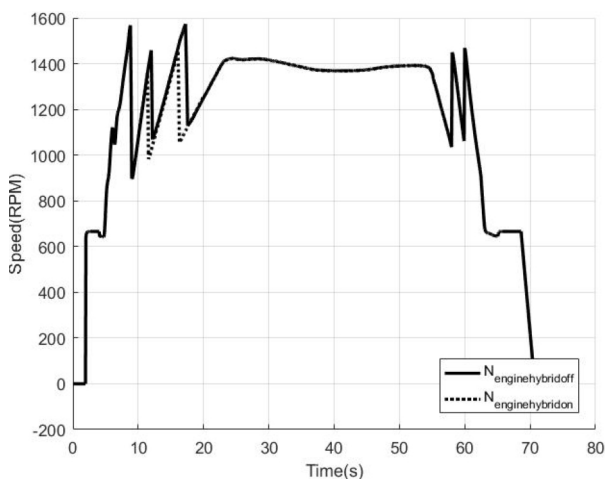


Figure 22. Hybrid drivetrain engine speed plot

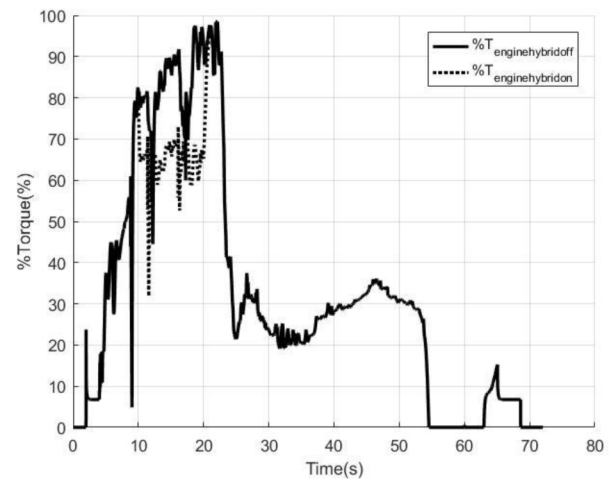


Figure 23. Hybrid drivetrain engine torque plot

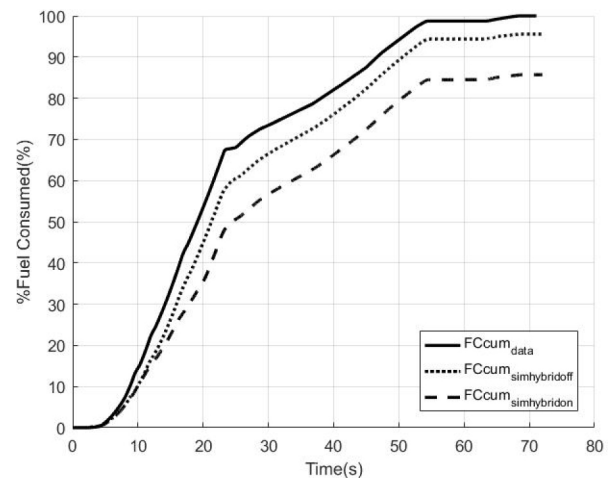


Figure 24. Fuel usage comparison for simulations vs real world driving data

The effect of the torque assist during the acceleration event is to reduce fuel consumption by an average of 7.7% compared to the previous simulation results with no hybridizing components. Also note that the velocity profile remains almost unchanged. However, the engine torque plot is noticeably different. When the torque assist is present during the acceleration phase, the engine outputs between 10% and 30% less torque throughout the 10 seconds when the assist is present and then returns to normal when the assist is removed. The additional torque source results in a fuel saving of 9.9% at the end of the simulation in comparison with the simulated conventional powertrain vehicle results. This represents a theoretical fuel saving as predicted by the model and demonstrates the models potential for identifying fuel saving opportunities within future concepts.

CONCLUSIONS

The proposed vehicle model architecture improves modelling flexibility by standardizing and replicating the real physical coupling between model subsystems. This allows for rapid reconfiguration of the model as well as testing and validation of individual component subsystems in isolation. The addition of an external torque source represented the addition of a hybridizing component to the drivetrain. Due to the new VMA, the external torque source was easily

incorporated without excessive modification to the existing drivetrain. It has been shown that the proposed modelling methodology is suitable for the simulation of vehicle dynamics closely matching real world driving data. The new VMA can also include complex components such as clutches and torque converters, accurately accounting for coupling and decoupling within the powertrain system. The use of the Simulink environment rather than a commercial package also benefits the user/developer by giving extra freedom in model development that other commercial packages may not give. Further work is to be completed on full drive cycle analysis using the new flexible architecture. Full validation of the model will also be performed to both conventional and hybrid drivetrain configurations. Development will also be conducted on the hybrid component models such as motor generator units and energy storage systems to improve the accuracy of the input data to the model whilst allowing additional results to be extracted.

REFERENCES

1. Murtagh, M., Kee, R., McCullough, G., Stuart, C., "Development and Validation of a Forklift Truck Powertrain Simulation," *SAE Int. J. Commer. Veh.* 6(1):150–160, 2013, doi:[10.4271/2013-01-0817](https://doi.org/10.4271/2013-01-0817).
2. Guzzella, L., and Sciarretta, A., "Vehicle Propulsion Systems: Introduction to Modeling and Optimization," Berlin: Springer, Berlin, ISBN:978-3-540-74691-1, 2007.
3. Belton, C., Bennett, P., Burchill, P., Copp, D., "A Vehicle Model Architecture for Vehicle System Control Design," SAE Technical Paper 2003-01-0092, 2003, doi:[10.4271/2003-01-0092](https://doi.org/10.4271/2003-01-0092).
4. Mason, B.A., Ebrahimi, M., Farid, M., "Reconfigurable modelling for drivetrain real-time simulation," *Proc. Inst. Mech. Eng. Part K J. Multi-Body Dyn.* 2009, 223, 309–323, doi:[10.1243/14644193JMBD191](https://doi.org/10.1243/14644193JMBD191).
5. Emirler, M.T., Uygan, I.M.C., Koprubasi, K., Unver, B., "Modular Heavy Duty Vehicle Modelling and User Interface Development" *IFAC PapersOnLine* 48–15(2015)354–360, doi:[10.1016/j.ifacol.2015.10.051](https://doi.org/10.1016/j.ifacol.2015.10.051).
6. Wipke, K.B., Cuddy, M.R., Burch, S.D., "ADVISOR 2.1: A User-Friendly Advanced Powertrain Simulation Using a Combined Backward/Forward Approach" *IEEE Transactions on Vehicular Technology* 48, (August):1751–1761, doi:[10.1109/25.806767](https://doi.org/10.1109/25.806767).
7. Milano, F., "An Open Source Power System Analysis Toolbox", *IEEE Transactions on Power Systems*, Vol. 20, No. 3, August 2005, doi:[10.1109/TPWRS.2005.851911](https://doi.org/10.1109/TPWRS.2005.851911).
8. Mathworks, "Building a Clutch Lock-Up Model", url: <http://uk.mathworks.com/help/simulink/examples/building-a-clutch-lock-up-model.html>
9. Eriksson, L., "Simulation of a vehicle in longitudinal motion with clutch lock and clutch release", *3rd IFAC workshop on Advances in automotive control*, 2001
10. Quang, H. N., "Dynamic Modelling of Friction Clutches and Application of this Model in Simulation of Drive Systems", *Periodica Polytechnica Ser. Mech. Eng.* Vol. 42, No.2, PP. 169–181, 1998.
11. Serrarens, A., Dassen, M., Steinbuch, M., "Simulation of an automotive dry clutch", *Proceedings of the American Control Conference*, 2004, doi:[10.1109/ACC.2004.182588](https://doi.org/10.1109/ACC.2004.182588)
12. Chintala, A., Narayana, R.L., Srinivas, C., "Modelling, Simulation and Control of an automotive clutch system", *International Journal of Engineering Research and Technology*, Vol. 2, Issue 12, 2013, ISSN:2278-0181
13. Bataus, M., Maciac, A., Oprean, M., Vasiliu, N., "Automotive clutch models for real time simulation", *Proceedings of the Romanian Academy*, Vol. 12, Issue 2, PP 109–116, 2011.

CONTACT INFORMATION

Gary Stevens
Queen's University Belfast
School of Mechanical and Aerospace Engineering
Ashby Building
Stranmillis Road
Belfast
BT9 5AG
gstevens01@qub.ac.uk

ACKNOWLEDGMENTS

The author would like to thank the Northern Ireland Department for Employment and Learning (DEL) and the Wrights Group Ltd for funding the research.

The author would also like to thank the Wrights Group Ltd for their technical contributions to the work and for the supply of the real world driving test data.